REMARKS

This communication is responsive to the Final Office Action dated September 8, 2005. Claims 1-53 are pending.

Claim Rejection Under 35 U.S.C. § 102

In the Final Office Action, the Examiner rejected claims 1-8, 11, 45-47, 51 and 53 under 35 U.S.C. 102(e) as being anticipated by Savage (USPN 6,442,687). Applicants respectfully traverse the rejection. Savage fails to disclose each and every feature of the claimed invention, as required by 35 U.S.C. 102(e), and provides no teaching that would have suggested the desirability of modification to include such features.

Before addressing the individual claims, Applicants provide some general comment to help the Examiner understand the differences between the claims and the cited art. In general, Applicants' claims are directed to an intermediate acceleration device that provides encryption and decryption services for a server. The intermediate acceleration device supports both a "direct mode" and a "full proxy mode." In either mode, the intermediate device manages secure communications with one or more clients, such as via SSL. However, the "direct mode" and "full proxy mode" differ by which type of communication session is used to forward unencrypted data packets from the intermediate device to the server. In either case, the intermediate device manages session negotiations between the client and the server and decrypts secure packets received from the client. In the "full proxy mode," the intermediate device forwards the decrypted data to the server using a communication session negotiated by the intermediate device. In other words, the intermediate device uses a different communication session than the session through which the encrypted data was received from the client. In "full proxy mode" the acceleration device handles both the SSL session and the TCP communications with the client. As illustrated in FIG. 7, the acceleration device negotiates both the SSL session and the TCP session with the client device. The intermediate acceleration device also negotiates a different TCP communication session with the server. In this manner, the intermediate acceleration device operates as a termination point for both TCP communication sessions, and handles copying data between communication sessions.

However, in the direct mode, <u>the intermediate device</u> still provides encryption and decryption services, but uses the <u>same</u> communication session that the client and sever negotiated. For example, the intermediate acceleration device intercepts secure communications from the client, provides decryption services to decrypt the data and forwards the decrypted data to the server using the *same* communication session. In this manner, only a single communication session is needed, and the communication session "cuts through" the intermediate device. This allows the intermediate acceleration device to transparently provide encryption and decryption services to secure communications without requiring the overhead and delay required by a full proxy mode. The present application describes the direct mode, also referred to as a "cut through mode," as follows:

> Figure 5 illustrates a direct, cut through processing method. Packets from client to server are addressed from the client to the server and from server to client, with the intermediary, SSL device being transparent to both. In the embodiment shown therein, the <u>SSL accelerator</u> allows the client and server to negotiate the TCP/IP session directly, making only minor changes to the TCP/IP headers passing through the accelerator device, and tracking session data in a data structure in memory to enable SSL session handling to occur. As described herein, this mode is referred to herein as the "direct, cut-through" mode, since the client and server "think" they are communicating directly with each other, and the SSL accelerator is essentially transparent.

As illustrated in FIG. 5, client 100 and server 300 negotiate the TCP/IP session and operate as termination points for the session. In other words, in "cut through" (direct) mode, the intermediate acceleration device still handles functions required for the SSL session, while the client and the server handle the TCP/IP session.

Savage (U.S. 6,442,687)

Savage describes a network system that uses two intermediate devices (an identity server and an action server) to separate a user's actions from its identity. Specifically, a client device separates an HTTP request into identity information and action information, which are encrypted at the client device.[1] The client device transmits the encrypted action and identity information to the identity server, which in turn forwards the action information to the action server. The identity server decrypts the identity information, and the action server decrypts the action information and forwards it to a third-party server. The third-party server sends the HTTP response back to the action server, which receives and encrypts the response, and forwards it to

the identity server. Finally, the identity server, which has been holding the unencrypted user identity information, receives the encrypted action response (which it cannot decipher), and forwards it to back to the client. This process is illustrated in FIG. 2 of Savage:

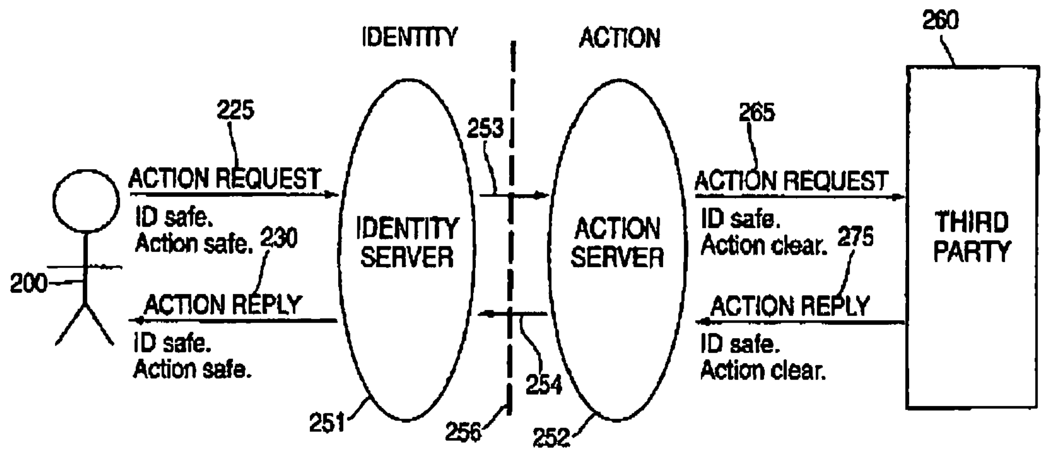

FIG. 2

The Savage system includes three major components: (1) a Java applet client 606 that runs on the client device and acts as an HTTP proxy for a user's web browser software, (2) identity server 251, and (3) action server 252 that provides another proxy server for forwarding action requests to the third party servers.

Savage states that once the identity server has received an HTTP request from the client, a separate connection is established between the identity server and an action server residing on a different physical computer.[2] This separate connection is used to forward the HTTP request from the identity server to the action server where it is decrypted. After decryption, action server 252 then forwards the clear text HTTP request to another HTTP proxy server that retrieves the requested URL and returns it to the action server. Savage makes clear that action server 252 decrypts action information for the HTTP request and forwards the decrypted information through another intermediate HTTP proxy to the destination third party server 260. Thus, separate communication sessions are maintained between the client and the identity server 251, between the identity server and the action server 252, and between the action server and the third

[1] Abstract.
[2] Savage at col. 8, ll. 35-55.

party server 260. *None of the intermediate devices of the Savage system support a direct mode in which the intermediate device decrypts data and forwards the unencrypted data packets from the intermediate device to the server using a communication session negotiated by the client and the server.* In contrast, all of the intermediate devices of the Savage system (i.e., the identity server and the action server) negotiate new communication sessions and use those sessions to forward data. None of the devices are able to transparently provide decryption services by essentially tapping into a communication session that was negotiated *by the client and the server* to forward data.

Claims 1-8, 11, 45-47, 51 and 53

Applicants' claim 1 requires managing a communications negotiation between the client and the server through an intermediate device that supports both a direct mode and a proxy mode. Further, claim 1 requires decrypting encrypted data packets, and forwarding unencrypted data packets from the intermediate device to the server *using a communication session negotiated by the client and the server when the intermediate device operates in direct mode.* Claim 1 further requires forwarding unencrypted data packets from the intermediate device to the server *using a second communication session negotiated by the server and the intermediate device* when the intermediate device operates in proxy mode.

In contrast, the intermediate devices of Savage only support a proxy mode that separately negotiates communication sessions for forwarding data.[3] As discussed above, Savage makes clear that separate communication sessions are maintained between the client and the identity server 251, between the identity server and the action server 252, and between the action server and the third party server 260. Savage does not describe an intermediate device that support a proxy mode in combination with a direct mode in which the intermediate device decrypts data from a client device and then forwards the unencrypted data packets from the intermediate device to the server using a communication session negotiated by the client and the server. In contrast, all of the intermediate devices of the Savage system (i.e., the identity server and the action

[3] It is important for the Office to understand that proxy servers establish separate communication sessions for forwarding packets. For example, a proxy server maintains a front-side communication session for communication with the client and a back-side session for communication with a server. See, e.g., www. pcmag.com,

server) negotiate new communication sessions and use those sessions to forward data. None of the devices are able to transparently provide decryption services by essentially tapping into a communication session that was negotiated by the client and the server to forward data.

With respect to the requirements of claim 1 relating to the direct mode, the Examiner cited col. 7, line 32 to col. 8, line 7. Applicants submit that this portion of Savage fails to describe or suggest an intermediate device that supports direct mode of acceleration at all. To the contrary, this portion only refers to a "proxy client" that, as explained above, requires separate communication sessions. For example, this portion of Savage (reproduced below) makes clear that the proxy "establishes" a separate communication session for forwarding the data and does not transparently use a session originally negotiated by the client device and the server. This portion makes clear that the proxy itself negotiates a new communication session. Thus, contrary to the Examiner's understanding, this portion of Savage does not describe or suggest an intermediate device that supports a direct mode of acceleration as claimed by the Applicant. Further, Applicants point out that the proxy client 606 described by Savage does not even execute on an intermediate device. To the contrary, the proxy client executes on the client device. Applicants refer the Examiner to FIG. 6 of Savage.

> *Proxy Client*
>
> *The proxy client of the preferred embodiment, a small footprint java applet 606, is the system component responsible for connecting end-users to the system. It functions as an HTTP proxy server and service HTTP requests from a user's web browser. Requests transferred through the system proxy client are encrypted and transferred to the identity server. Responses received by the proxy client from the action server via the identity server are decrypted and returned to a user's web browser.*
>
> *Upon invocation from a known URL on the world-wide-web, the proxy client is loaded from a JAR file by a client web browser. Once loaded, the proxy client generates and/or retrieve the cryptographic data required <u>to establish a secure communication channel with the system action server</u>, and automatically configures the user's web browser to use the proxy client as a proxy server for browsing the world-wide-web (or alternately prompts the user to make this setting manually).*
>
> *After receiving an HTTP request generated by a user's web browser, <u>the proxy client establishes a secure connection to the identity server</u> using the communication protocol discussed later in this disclosure.*[4]

---

www.computing-dictionary.com stating that a proxy server is "a computer system or router that **breaks the connection** between sender and receiver." (emphasis added).

[4] Savage at col. 7, line 32 to col. 8, line 7 (emphasis added).

Claim 45 requires an SSL acceleration device having a communication engine that supports: (1) *a direct mode in which the intermediate SSL acceleration device decrypts data packets and forwards the decrypted data to the servers using a communication session negotiated by the client and the server*, and (2) a proxy mode in which the acceleration device responds to the client on behalf of the server and forwards the decrypted data packets to the server using the open communications session established by the acceleration device and the server. Savage does not describe an SSL acceleration device that support a proxy mode in combination with a direct mode as claimed by the Applicants.

For at least these reasons, Savage fails to teach or suggest the requirements of independent claims 1, 33 and 45. Moreover, none of the other references, either singularly or in combination, provide any teaching or suggests that overcome the deficiencies of Savage.

With respect to dependent claims 51 and 53, the Examiner asserted that Savage teaches automatically switching the intermediate device from the direct mode to the proxy mode upon detecting a communication error associated with the direct mode. In rejecting claims 51 and 53, the Examiner cited col. 7, ln. 32 to col. 8, ln. 7, reproduced above. As discussed above, the portion of Savage solely describes a proxy mode and does not teach or suggest an intermediate device that supports a direct acceleration mode at all. Moreover, the Applicants are at a loss as to where the Examiner finds a teaching for an intermediate device that automatically switches from the direct acceleration mode to the proxy mode upon detecting a communication error associated with the direct mode. The portion of Savage cited by the Examiner does not even mention communications errors or switching between modes. Savage only states that if the system fails to respond to a proxy client's request for a specified time-out interval, the proxy client aborts request processing and returns an error page to the user's web browser. This does not refer to switching acceleration modes within an intermediate device, only that an error page is returned to the web browser. The proxy client does not switch to a proxy mode from a direct mode, as required by Applicants' claims. The proxy client always operates as a proxy and never switches to a proxy mode from any other type of mode. Finally, the Applicants again point out that the proxy client relied upon by the Examiner executes on the client device and not within an intermediate device at all. Thus, the device on which the proxy client executes (i.e., the client device) is not performing any form of acceleration and does not decrypt packets received from a

client, as required by Applicants' claims. To the contrary, the proxy client essentially is the client device (or at least operates therein).

Savage fails to disclose each and every limitation set forth in claims 1-8, 11, 45-47, 51 and 53. For at least these reasons, the Examiner has failed to establish a prima facie case for anticipation of Applicants' claims 1-8, 11, 45-47, 51 and 53 under 35 U.S.C. 102(e). Withdrawal of this rejection is requested.

Claim Rejection Under 35 U.S.C. § 103

In the Final Office Action, the Examiner rejected claims 9, 10, 12-44, 48-50 and 52 under 35 U.S.C. 103(a) as being unpatentable over Bellwood (USPN 6,584,567) or Savage in various combinations with other references. Applicants respectfully traverse the rejection. The applied references fail to disclose or suggest the inventions defined by Applicants' claims, and provide no teaching that would have suggested the desirability of modification to arrive at the claimed invention.

In rejecting independent claim 20, the Examiner primarily relied on Bellwood. Applicants' independent claim 20 recites steps performed by an intermediary device when using the "direct" (cut through) communication session described above. For example, claim 20 requires establishing a communications session between the client and said one of said plurality of servers by receiving negotiation data from the client intended for the server and forwarding the negotiation data in modified form to the server, and receiving negotiation data from the server intended for the client and forwarding the negotiation data to the client to establish the client and the server as terminations for the communications session. In this manner, claim 20 requires that a communication session is established through the intermediary device and terminated by the client and the server. In addition, claim 20 requires establishing a secure communications session between the client and the intermediary device, and forwarding decrypted application data from the intermediary device to said one of said plurality of servers using the communications session established between the client and the server.

With respect to Bellwood, it appears the Examiner has again misunderstood Applicants' claim elements. In contrast to these elements of Applicants' claim 20, Bellwood describes a method of enabling a proxy to participate in a secure communication between a client and a set

of servers. The method begins by establishing **<u>a first secure session between the client and the proxy</u>**. Upon verifying the first secure session, the method continues by establishing a **<u>second secure session between the client and the proxy</u>**.[5] Thus, Bellwood clearly describes using two separate communication sessions, and makes quite clear that a separate communication session is established by the proxy and the server for forwarding data from the proxy to the server. This is in direct contrast with Applicants' claim 20. Bellwood is not describing a direct acceleration method in which a communication session is established through the intermediate acceleration device and <u>terminated by client and the server</u>. Bellwood fails to describe the intermediate device forwarding decrypted application data from the intermediary device to said one of said plurality of servers *using the communications session established between the client and the server*, as required by claim 20.

In rejecting independent claim 33, the Examiner primarily relied on Savage. Applicants' independent claim 33 requires an acceleration apparatus adapted to operate in either one of a direct mode and a proxy mode. Claim 33 requires that in the direct mode the intermediate *acceleration apparatus decrypts data packets received from the client and forwards the decrypted data packets to one of the servers using a communication session negotiated by the client and the server*, and in the proxy mode the acceleration apparatus responds to the client on behalf of the server and forwards the decrypted data packets to the server using a communication session negotiated by the acceleration device and the server. For reasons set forth above, Savage does not describe an acceleration apparatus that support a proxy mode in combination with a direct mode as claimed by the Applicants.

With respect to Applicants' dependent claims, none of the other references, either singularly or in combination, address this basic deficiency of the prior art with respect to an intermediate acceleration device that combines a full proxy mode with a direct mode. None of the references, either singularly or in combination, teach or suggest an intermediate device in which <u>the intermediate device</u> transparently provides encryption and decryption services, but uses the <u>same</u> communication session that the client and server originally negotiated to forward decrypted data to the server. For example, Cohen describes a proxy architecture that uses separate TCP connections and fails to describe a direct or cut through mode. Maloney et al.

---

[5] Abstract.

describes an information analysis system that is a combination of sensor, analysis, data conversion, and visualization programs. Boeuf describes a file server that stores data by allocating a single oversized contiguous storage area and by allowing data wrapping. Fujiyama et al. describes a network system in which each of multiple networks, each containing computers and relay computers, is connected to another network via multiple relay computers. None of the relay computers act as acceleration devices. Bellaton et al. describes a mechanism for dispatching a sequence of packets via a telecommunications network. Gelman et al. describes a method of communicating over a satellite or other high delay-bandwidth link that does not utilize TCP/IP. Holtey et al. describes a secure memory card and is unrelated to a network acceleration device. Harper et al. describes techniques for rejuvenating a component of a distributed data processing environment.

The Examiner has failed to establish a prima facie case for non-patentability of Applicants' claims 9, 10, 12-44, 48-50 and 52 under 35 U.S.C. 103(a). Withdrawal of this rejection is requested.

CONCLUSION

All claims in this application are in condition for allowance. Applicants respectfully request reconsideration and prompt allowance of all pending claims. Please charge any additional fees or credit any overpayment to deposit account number 50-1778. The Examiner is invited to telephone the below-signed attorney to discuss this application.

Date: November 8, 2005

By: Kent J. Sieffert

SHUMAKER & SIEFFERT, P.A.
8425 Seasons Parkway, Suite 105
St. Paul, Minnesota 55125
Telephone: 651.735.1100
Facsimile: 651.735.1102

Name: Kent J. Sieffert
Reg. No.: 41,312